Collabits Journal
Vol 1 No. 2 May 2024 : 77-82
E-ISSN : 3046-6709, P-ISSN : 3062-8601
https://publikasi.mercubuana.ac.id/index.php/collabits

Installing the Haiku Operating System on Virtual Box and Implementing File Management

Mohamad Yusuf[1*], M Reza Al Fatah[2], Asrul Fami[3], Vemas Adi Pratama[4], M Fauzan Z[5], Muhammad Syadam[6]

[1,2,3,4,5,6] Informatics Engineering Study Program, Universitas Mercu Buana. Indonesia

*Coressponden Author: mhd.yusuf@mercubuana.ac.id

Abstract - This journal aims to explore the installation process of the Haiku operating system, utilizing VirtualBox and implementing effective file management techniques. Haiku, as an open-source operating system, showcases an innovative user interface and high performance. The study assesses the practicality of Haiku in virtualization through the use of the VirtualBox application. The journal presents step-by-step instructions on installing the Haiku operating system and deploying it within the VirtualBox application. Additionally, it delves into various aspects of file management in the Haiku system, including creation, deletion, and data viewing within files.

Keywords :
Haiku;
Operating system;
VirtualBox;
Manajemen File;

Article History:
Received: 15-02-2024
Revised: 21-03-2024
Accepted: 06-04-2024

Article DOI : 10.22441/collabits.v1i2.26054

1. INTRODUCTION

Haiku is a breakthrough operating system designed to replace BeOS (Be Operating System) with modern innovation and efficiency. Has the foundation to support speed, flexibility, and user interface. Haiku itself provides a compelling experience for users looking for a lightweight operating environment. Haiku's file structure is inspired by BeFS (Be File System), increasing data access speed and storage capacity. With efficient graphics processing and flexible memory management, Haiku provides a powerful foundation for multimedia and creative applications.

The development history of Haiku begins with the completion of improvements to BeOS by Be, Inc. in early 2000. A passionate developer community formed the Haiku project to continue and develop the legacy of BeOS. Through important milestones, the Haiku project grew from ambition to a mature operating system.

Scheme for installing and experimenting with the Haiku operating system. This research uses VirtualBox, a well-known virtualization platform. VirtualBox allows users to create and manage virtual machines, creating an isolated space for testing operating systems without affecting the main operating system.

File management plays an important role in using the operating system. Therefore, this research investigates Haiku's unique approach to file management that simplifies the data management and access process for users.

2. LITERATURE REVIEW

Haiku is an alternative open source operating system that is still under development. Haiku, also known as OpenBeOs is an operating system created as a development of BeOS, an operating system with a powerful architecture originally developed by Be Inc. BeOS is differentiated from other operating systems because it was created to function on modern hardware and is not widely recognized because the project was discontinued after difficulties in commercialization efforts. Originally running on the PowerPC architecture, BeOS was later ported to the X86 platform towards the end of its life. BeOS has advantages, such as a strong architecture for multimedia, support for multiprocessors, and a 64-bit journal file system.

On the other hand, Haiku is a completely new project and has no direct connection to the BeOS source code. As an open source project, Haiku is still in the preAlpha stage, requiring virtual applications such as Qemu or Vmware to run. Although Haiku inherited some positive characteristics from BeOS, the project developed its own identity and development trajectory.

3. METHODOLOGY

In this research, the first step taken was to understand the kernel structure of the Haiku operating system, which was considered an initial and crucial step. This research journal provides a detailed explanation of the kernel

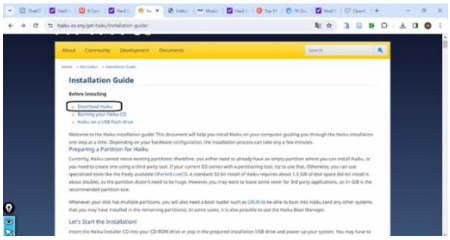
components in the Haiku operating system, presenting a comprehensive picture of how the systems interact with each other. This involves addressing fundamental tasks such as process management, memory allocation, and file operations, which are core elements of the Haiku operating system.

Steps for installing the Haiku operating system :

1. Visit the official website of haiku.

2. Select the install haiku menu then select the installation guide section.

3. Select the download haiku menu.

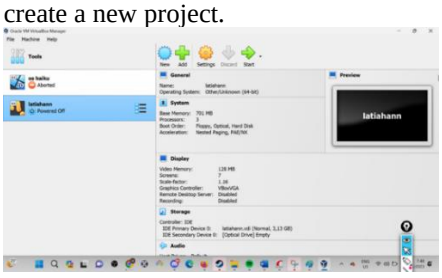


4. Select the New York location and select the bit according to your device.

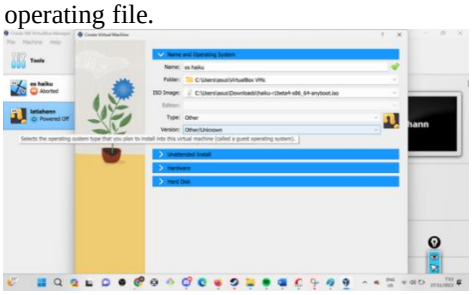
5. Virtual Box Installation.

Running the Haiku Operating System :

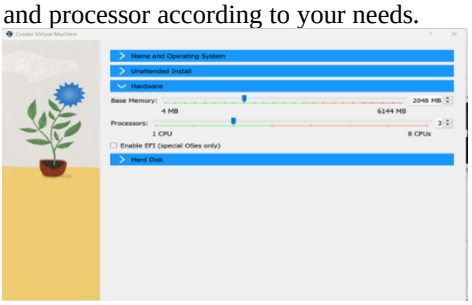
1. Open virtual box then select the new menu to create a new project.



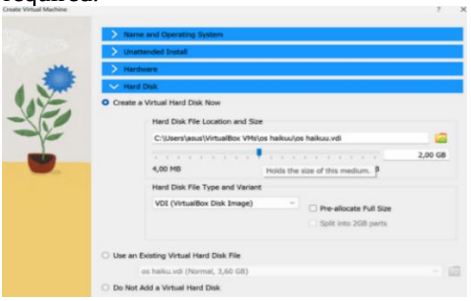
2. Create a project name then point to your project location and finally point the ISO image to the downloaded Haiku system operating file.



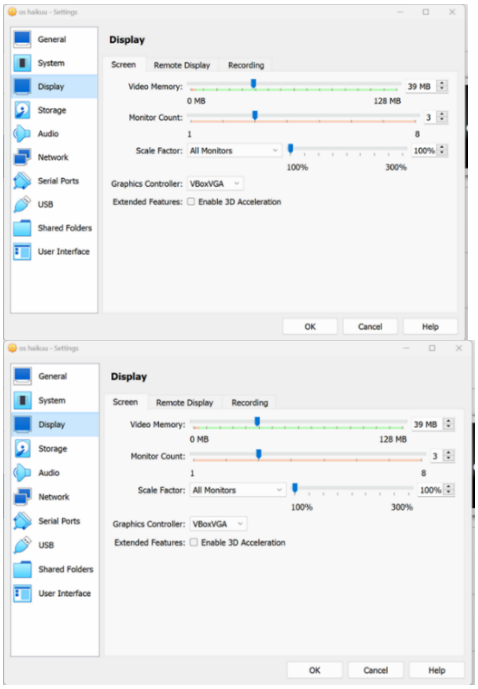
3. In the hardware section, set the base memory and processor according to your needs.



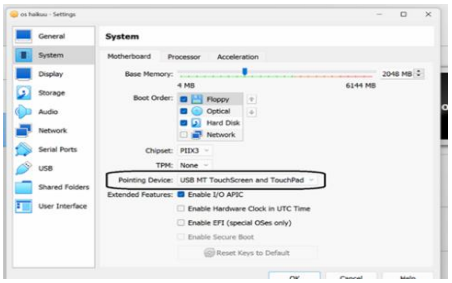
4. Then in the hard disk section, set the storage required.



5. Once created, set the display according to your needs.

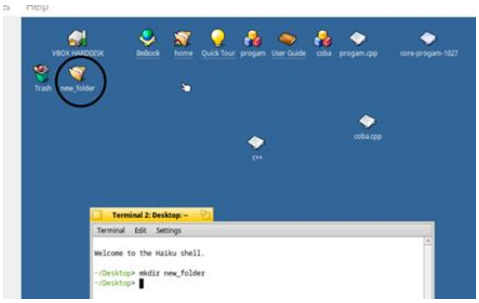


6. Select the system menu then set the pointing device so you can use the cursor in the Haiku OS and base memory as needed.



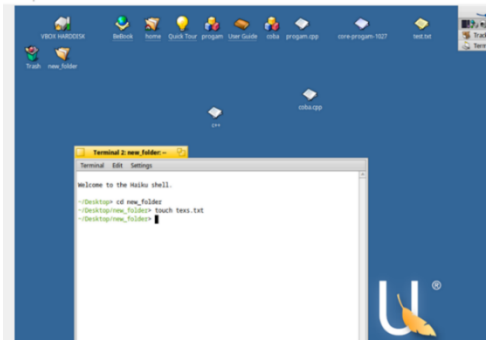
7. Click the start button to run Haiku OS.

1) File Management in Haiku with Terminal.
Using keyboard shortcuts in Haiku desktop, we can manage system files. Once we enter the terminal, we can create, rename, read, and delete files as well as add and delete directories. Create directory We can create a directory in Haiku in the following way.



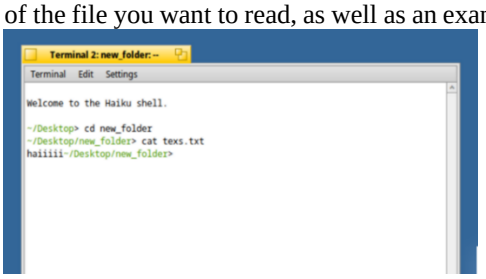
Then a new folder file has been created in the Haiku system.

2) Create a new file in the rectory that has been created.
The first step is the same as the previous step, namely opening the terminal, then changing the directory to the existing directory and creating a file in that directory.



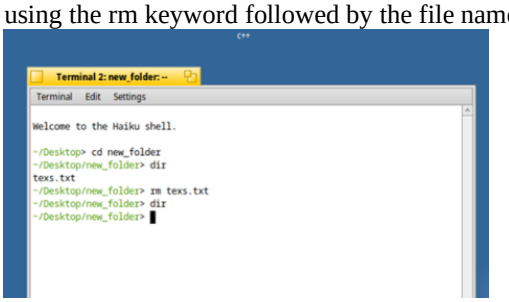
3) View the contents of the files in the directory.
Viewing files in a directory can be done by typing dir, here is an example of viewing files in a directory using the terminal on the Haiku operating system.

4) View the contents of the file data.
Reading files in the Haiku system operation can be done with the key word cat followed by the name of the file you want to read, as well as an example.



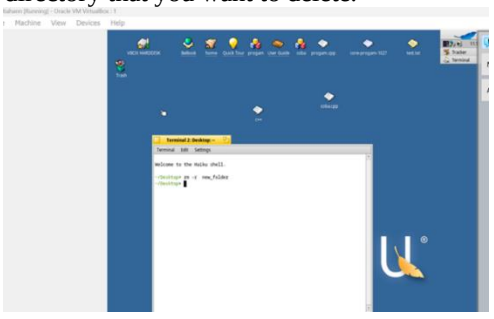
Open the haiku system then right click and create a file with the extension cpp to create a program using c++, example program .cpp

5) Deleting files. We can delete files in certain directories with the Haiku operating system by using the rm keyword followed by the file name.



In the image above, previously there was a text.txt file, but after running the rm keyword, when you returned to dir, the file data was not there.

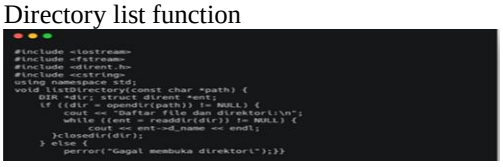
6) Director remover. To delete a directory, we can use the key word rm-r followed by the name of the directory that you want to delete.



Then the new folder director has been deleted File Management with the C++ Program in Haiku The first step is to right click on the desktop then click new and select text file and create a file in cpp format

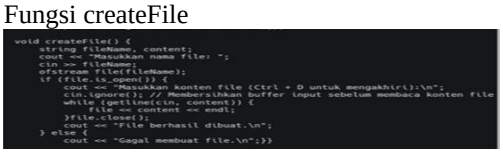
Write coding in the C++ files that have been created

1) Directory list function



- **Description**: This function opens the given directory and displays a list of files and directories in that directory.
- **Implementation**: Uses the opendir and readdir functions to read directory contents. Each item (file or directory) is displayed using ent-d_name.
- **Error Handling**: If the directory cannot be opened, an error message will be displayed using perror.

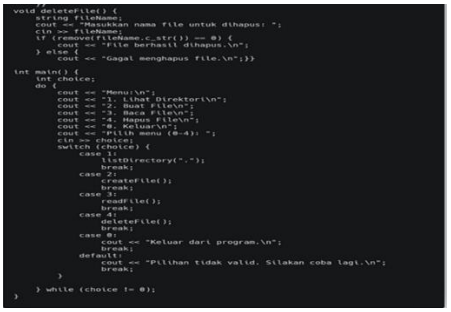
2) Fungsi createFile



- Description: This function prompts the user to enter a new file name and its contents. Then, create a file with that name and write the contents into it.
- Implementation: Use ofstream to create and open files. The file content is filled in by the user via cin.
- Important: cin.ignore() is used to clear the input buffer before reading the file contents so that it is not affected by the newline (\n) after entering the file name.

3) ReadFile function

- Description: This function prompts the user to enter the name of the file to be read. Then, displays the contents of the file on the screen.
- Implementation: Uses fstream to open the file and getline to read and display its contents.
- Error Handling: If the file cannot be opened, an error message will be displayed.



4) DeleteFile function

- Description: This function prompts the user to enter the name of the file to delete. Then, delete the file
- Implementation: Uses the remove function to delete files.
- Error Handling: If the file cannot be deleted, an error message will be displayed

To see system performance, you can use Benchmarks. Benchmarks are tests carried out to measure the performance of a system or hardware. Benchmarks are used using the defaults from Haiku, namely top and vmstat. Top Benchmarks are benchmarks that measure the performance of the operating system. This benchmark uses various tests to measure operating system performance in various aspects, such as CPU performance and examples of using top benchmarks.

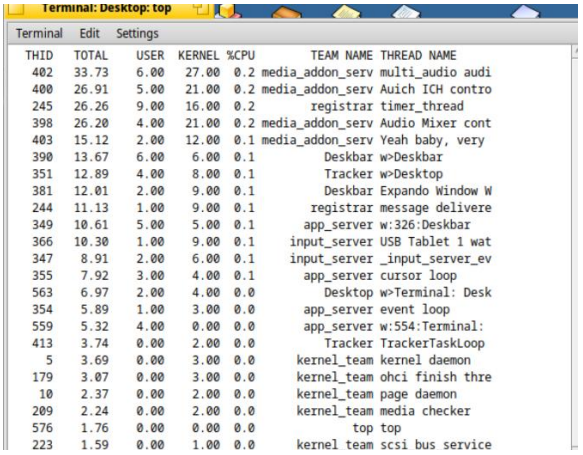


Figure 1. Haiku OS performance

The following is a brief explanation of the information image:

- Thid
  Thid is an abbreviation of Thread. A thread is the basic work unit of a process. Each process can have one or more threads. Threads can be used to divide heavy tasks into smaller, more manageable tasks.
  In the top view, third shows the number of threads currently running on the system. You can use this information to determine how busy your system is.
- Totals
  Total is the total number of threads currently running on the system. This information can be used to determine the number of threads currently

running on the system.

- Users
  User is the number of threads currently running in user mode. User mode is the mode in which the process is run by the user. This information can be used to determine the number of threads currently running by the user.
- Kernels
  Kernel is the number of threads currently running in kernel mode. Kernel mode is the mode in which processes are run by the kernel. This information can be used to determine the number of threads the kernel is running.
- CPU
  CPU is the number of CPUs used by all threads. This information can be used to determine the amount of CPU used by the system

Next, use the vmstat(7) benchmark, vmstat is a computer system monitoring tool that collects and displays summary information about operating system memory, processes, interrupts, paging, usage examples:

```
max memory:             733937664
free memory:            634748928
needed memory:          0
block cache memory:     2621440
max swap space:         612634624
free swap space:        612634624
page faults:            124134

page faults  used memory    used swap  block cache
        428             0           0            0
         22             0           0            0
          3             0           0            0
          1             0           0            0
          0             0           0            0
          0             0           0            0
          2             0           0            0
          0             0           0            0
          0             0           0            0
          0             0           0            0
          8             0           0            0
```

The following is a brief explanation of the information in the image:

- Max memory
  Max memory is the maximum amount of physical memory available on the system. This information can be used to determine how much physical memory is available for running processes.
- Free memory
  Free memory is the amount of physical memory that is not currently in use. This information can be used to determine how much physical memory is available for running processes.
- Required memory
  Needed memory is the amount of physical memory required by running processes. This information can be used to determine whether the system has enough physical memory to run the processes currently running.
- Block cache memory
  Block cache memory is the amount of physical memory used to store disk block data. This information can be used to determine how much physical memory to use to store frequently used data.
- Max swap space
  Max swap space is the maximum amount of swap space available on the system. Swap space is secondary storage space used to store data that cannot be stored in physical memory.
- Free swap space
  Free swap space is the amount of swap space that is not used at this time. This information can be used to determine how much swap space is available to store data that cannot be stored in physical memory.
- Page results
  Page result is the number of pages moved to or from virtual memory. This information can be used to determine how actively the system is moving data to and from physical memory.
- Used memory
  Used memory is the amount of physical memory used by running processes, including memory used to store disk block data. This information can be used to determine how much physical memory is used by the system.
- Used swap
  Used swap is the amount of swap space used by running processes. This information can be used to determine how much swap space is used by the system.
- Block cache
  Block cache is the amount of physical memory used to store frequently used disk block data. This information can be used to determine how effectively the system uses swap space.[7]

4. RESULTS AND DISCUSSION

In the context of benefits, Haiku stands out with its responsive and intuitive user interface. Focus on simple yet elegant design provides a pleasant and efficient user experience. Another advantage lies in its high performance and responsiveness, especially when used in a professional environment. Haiku's ability to perform well, even in tiring tasks, provides users with confidence and comfort.

Furthermore, ever-increasing hardware compatibility is an added advantage. Haiku continues to expand its support for a variety of hardware, giving users the flexibility to adopt it across multiple platforms. Increased hardware compliance is a positive boost that enriches the user experience with a wide variety of hardware applications.

On the other hand, Haiku's main weakness is its limitations in the number and quality of third-party applications. Although there are efforts to develop Haiku applications, these limitations may hinder users who require various specialized applications. Another problem lies in the lack of strong commercial support. These support limitations may impact Haiku's appeal in business and industrial sectors that require assurance of ongoing support and updates.

The final point covers the learning curve that may be

higher for users new to Haiku. The process of adapting to Haiku's unique interface and concept requires time and in-depth knowledge. This can be a drag, especially for users used to other operating systems. Despite its high learning potential, with further understanding of the Haiku interface, users can overcome this barrier

5. CONCLUSION

At the end of this research, it was found that the Haiku operating system provided efficient and optimal results. Responsive interface, compatibility for complex systems, and superior features make Haiku highly recommended for creative use. File management is one of the superior features of this operating system, apart from that, the responsive performance and ease of use of the Haiku operating system makes it the choice of choice for all groups, both laypeople and professionals.

With all the advantages of the Haiku operating system, it cannot be denied that this operating system still has various challenges to overcome. Lack of commercial support may indicate the lack of popularity of this operating system. Apart from that, the onslaught of new innovations in other operating systems means that Haiku must continue to improve its ecosystem.

Haiku's potential as an alternative operating system in the development of creative applications and its use as a backup technology in an operating system environment. With deeper understanding, Haiku can become a relevant solution in meeting the needs of growing industries.

Therefore, an alternative open source operating system, Haiku, has potential for further development. Strengthening commercial support is also needed to support Haiku's business growth. In addition, it improves training for its users and performance for development as an operating system.

Combining this research and recommendations for further development, this research provides deep insight into the potential and challenges of Haiku as an alternative open source operating system. Haiku has the opportunity to grow and find its place in the operating system market by encouraging users to be creative and appropriate development strategies.

REFERENCE

[1] Аникин, К. Э., & Ефремова, Д. П. (2016). Alternative operating systems. Язык в сфере профессиональной коммуникации.—Екатеринбург, 2016, 246-250. https://elar.urfu.ru/bitstream/10995/86233/1/978-5-8295 0435-9_2016_061.pdf

[2] Leavengood, R. (2012). The Dawn of Haiku-How a volunteer crew brought a crack OS back. IEEE Spectrum, 49(5), 40-54. https://ieeexplore.ieee.org/abstract/document/6189574/

[3] Robinson, R. (2005). BEOS, THE BE OPERATING SYSTEM. CS-550-1: OPERATING SYSTEMS. FALL 2005.

[4] Dolan-Gavitt, B., Leek, T., Zhivich, M., Giffin, J., & Lee, W. (2011, May). Virtuoso: Narrowing the semantic gap in virtual machine introspection. In 2011 IEEE Symposium on Security and Privacy (pp. 297-312). IEEE.

[5] Revol, F. (2011, September). Universal file system extended attributes namespace. In International Conference on Dublin Core and Metadata Applications (pp. 69-73).

[6] Revol, F. (2011, March). Showing and Debugging Haiku with QEMU. In 1st International QEMU Users' Forum (p. 31).

[7] CodeAhoy [https://codeahoy.com/compare/top-vs-vmstat]. 2024 [cited 2024 Mar 1]. p. 1–1 Compare top and vmstat.